

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-301773

(43) 公開日 平成10年(1998)11月13日

(51) Int.Cl.⁹

G 0 6 F 9/06

識別記号

5 5 0

F I

G 0 6 F 9/06

5 5 0 A

5 5 0 E

5 5 0 Z

審査請求 未請求 請求項の数16 O L (全 22 頁)

(21) 出願番号

特願平9-112182

(22) 出願日

平成9年(1997)4月30日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 竹内 彰一

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 難波 慎二

東京都品川区北品川6丁目7番35号 ソニー株式会社内

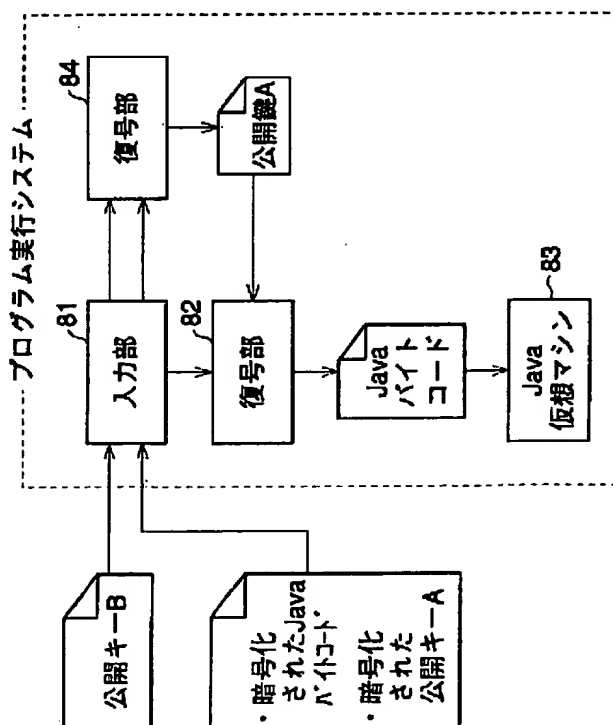
(74) 代理人 弁理士 稲本 義雄

(54) 【発明の名称】 情報処理装置および情報処理方法、並びに記録媒体

(57) 【要約】

【課題】 あるプログラム実行環境において、正当なソフトウェア開発者が開発したプログラムのみが実行されるようにする。

【解決手段】 プログラム実行システムにおいて、秘密キーAを用いて暗号化されたアプリケーションプログラムは復号部82に供給され、公開キーB、およびその公開キーBに対応する秘密キーBを用いて暗号化された公開キーA（秘密キーAに対応するもの）は復号部84に供給される。復号部84では、暗号化された公開キーAが、公開キーBを用いて復号化され、復号部82に供給される。復号部82では、復号部84からの公開キーAを用いて、暗号化されたアプリケーションプログラムが復号化され、その復号結果としてのJavaバイトコードが、Java仮想マシン83に供給される。Java仮想マシン83では、復号部82からのJavaバイトコードが解釈、実行される。



【特許請求の範囲】

【請求項1】 プログラムを実行するための処理を行う情報処理装置であって、

前記プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化する第1のキー復号化手段と、

前記第1のキー復号化手段により復号化された前記第1のキーを用いて、前記プログラムを暗号化したものを復号化するプログラム復号化手段と、

前記プログラム復号化手段が出力する前記プログラムを実行する実行手段とを備えることを特徴とする情報処理装置。

【請求項2】 前記第2のキーが暗号化されている場合において、

前記第2のキーを暗号化したものを、第3のキーを用いて復号化する第2のキー復号化手段をさらに備えることを特徴とする請求項1に記載の情報処理装置。

【請求項3】 前記第1のキーは、前記プログラムを共通鍵暗号化方式で暗号化するのに用いられた共通キーであり、

前記第2のキーは、前記第1のキーを公開鍵暗号化方式で暗号化するのに用いられた秘密キーに対応する公開キーであり、

前記第3のキーは、前記第2のキーを公開鍵暗号化方式で暗号化するのに用いられた秘密キーに対応する公開キーである。ことを特徴とする請求項2に記載の情報処理装置。

【請求項4】 前記プログラム復号化手段は、複数の前記第1のキーを用いて、前記プログラムを暗号化したものを復号化することを特徴とする請求項1に記載の情報処理装置。

【請求項5】 プログラムを実行するための処理を行う情報処理方法であって、

前記プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化し、

その復号化された前記第1のキーを用いて、前記プログラムを暗号化したものを復号化し、

その復号化された前記プログラムを実行することを特徴とする情報処理方法。

【請求項6】 コンピュータに、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化させ、

その復号化された前記第1のキーを用いて、前記プログラムを暗号化したものを復号化させ、

その復号化された前記プログラムを実行させるためのプログラムが記録されていることを特徴とする記録媒体。

【請求項7】 前記第2のキーも記録されていることを特徴とする請求項6に記載の記録媒体。

【請求項8】 プログラムを処理する情報処理装置であって、

請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段を備えることを特徴とする情報処理装置。

【請求項9】 プログラムを処理する情報処理方法であって、

請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化することを特徴とする情報処理方法。

【請求項10】 請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されていることを特徴とする記録媒体。

【請求項11】 プログラムを実行するための処理を行う情報処理装置であって、

前記プログラムを実行する実行手段と、

前記プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化するキー復号化手段と、

前記キー復号化手段により復号化された前記第1のキーを用いて、前記プログラムに付されている署名が正当なものかどうかを確認する確認手段と、

前記確認手段により正当なものであることが確認された署名が付されている前記プログラムを、前記実行手段に供給する供給手段とを備えることを特徴とする情報処理装置。

【請求項12】 プログラムを実行するための処理を行う情報処理方法であって、

前記プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化し、

その復号化された前記第1のキーを用いて、前記プログラムに付されている署名が正当なものかどうかを確認し、

前記署名が正当なものであることが確認された場合のみ、前記プログラムを実行することを特徴とする情報処理方法。

【請求項13】 コンピュータに、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化させ、

その復号化された前記第1のキーを用いて、前記プログラムに付されている署名が正当なものかどうかを確認させ、

前記署名が正当なものであることが確認された場合のみ、前記プログラムを実行させるためのプログラムが記録されていることを特徴とする記録媒体。

【請求項14】 プログラムを処理する情報処理装置であって、

請求項11に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理する処理手段を備えることを特徴とする情報処理装置。

【請求項15】 プログラムを処理する情報処理方法であって、

請求項11に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理することを特徴とする情報処理方法。

【請求項16】 請求項11に記載の情報処理装置において署名が正当なものであると確認されるように処理されたプログラムが記録されていることを特徴とする記録媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、情報処理装置および情報処理方法、並びに記録媒体に関し、特に、あるプログラム実行環境において、正当なソフトウェア開発者が開発したプログラムのみが実行されるようにする情報処理装置および情報処理方法、並びに記録媒体に関する。

【0002】

【従来の技術】最近急速に普及してきたインターネットに適していることから、Java（米国Sun Microsystems社の商標）が注目されている。Javaは、オブジェクト指向言語であるJava言語や、そのJava言語で記述されたプログラム（以下、適宜、Javaプログラムという）の実行に適したプロセッサのアーキテクチャを定義した仮想マシン（以下、適宜、Java仮想マシンという）、その他のJavaと関連する要素を呼ぶのに、あるいは、それらの総称として用いられる。なお、ここでいう仮想マシンとは、ユーザに対して1台のコンピュータを複数のコンピュータに仮想的に見せる場合の仮想マシンではなく、言語処理系を実装する場合に想定する仮想的なマシンを意味する。

【0003】Java仮想マシンは、実際の（現実の）コンピュータ上において、そのコンピュータにインストールされているOS（Operating System）上で動作するように実装される。一方、Javaプログラムは、Java仮想マシンの命令セットからなるバイナリコードにコンパイルされる。このバイナリコードは、Java仮想マシンが動作するどのようなハードウェアでも実行することができる。従って、Java仮想マシンさえ動作すれば、コンパイル済みのJavaプログラムは、種々のプラットフォームで実行することができる。

【0004】

【発明が解決しようとする課題】上述したように、Java仮想マシンを実装すれば、どのようなマシン上でも、Javaプログラムを実行することができることから、Java仮想マシンが、多数のユーザに普及することが予想され、さらに、そのような多数のユーザ向け

に、多くのアプリケーションプログラムが開発、配布（有償、無償を問わない）されることが予想される。

【0005】このような状況の下、Java仮想マシンその他のプログラム実行環境を開発、配布した者からすれば、自身が開発等したプログラム実行環境下において実行される、第三者により開発されたアプリケーションプログラムの配布を制限したい場合がある。即ち、例えば、ライセンス契約を結んだ者だけに、アプリケーションプログラムの配布を許可したい場合がある。

【0006】一方、Java仮想マシンでは、JavaコンパイラでJavaプログラムを、バイトコード（Javaバイトコード）と呼ばれる中間コードにコンパイルしたものが解釈されて実行されるが、Javaバイトコードは、それを逆コンパイルすることにより、比較的容易に理解することができるため、リバースエンジニアリングを簡単に行うことができる。従って、第三者による模倣や改竄を防止する必要がある。

【0007】本発明は、このような状況に鑑みてなされたものであり、あるプログラム実行環境下におけるプログラムの実行を制限することができるようにし、さらに、プログラムの模倣や改竄を防止することができるようにするものである。

【0008】

【課題を解決するための手段】請求項1に記載の情報処理装置は、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化する第1のキー復号化手段と、第1のキー復号化手段により復号化された第1のキーを用いて、プログラムを暗号化したものを復号化するプログラム復号化手段と、プログラム復号化手段が出力するプログラムを実行する実行手段とを備えることを特徴とする。

【0009】請求項5に記載の情報処理方法は、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化し、その復号化された第1のキーを用いて、プログラムを暗号化したものを復号化し、その復号化されたプログラムを実行することを特徴とする。

【0010】請求項6に記載の記録媒体は、コンピュータに、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化させ、その復号化された第1のキーを用いて、プログラムを暗号化したものを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されていることを特徴とする。

【0011】請求項8に記載の情報処理装置は、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段を備えることを特徴とする。

【0012】請求項9に記載の情報処理方法は、請求項1に記載の情報処理装置で実行可能なコードに復号化さ

れる暗号文に、プログラムを暗号化することを特徴とする。

【0013】請求項10に記載の記録媒体は、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されていることを特徴とする。

【0014】請求項11に記載の情報処理装置は、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化するキー復号化手段と、キー復号化手段により復号化された第1のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認する確認手段と、確認手段により正当なものであることが確認された署名が付されているプログラムを、実行手段に供給する供給手段とを備えることを特徴とする。

【0015】請求項12に記載の情報処理方法は、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化し、その復号化された第1のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認し、署名が正当なものであることが確認された場合のみ、プログラムを実行することを特徴とする。

【0016】請求項13に記載の記録媒体は、コンピュータに、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化させ、その復号化された第1のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認させ、署名が正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されていることを特徴とする。

【0017】請求項14に記載の情報処理装置は、請求項11に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理する処理手段を備えることを特徴とする。

【0018】請求項15に記載の情報処理方法は、請求項11に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理することを特徴とする。

【0019】請求項16に記載の記録媒体は、請求項11に記載の情報処理装置において署名が正当なものであると確認されるように処理されたプログラムが記録されていることを特徴とする。

【0020】請求項1に記載の情報処理装置において、第1のキー復号化手段は、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化し、プログラム復号化手段は、第1のキー復号化手段により復号化された第1のキーを用いて、プログラムを暗号化したものを復号化するようになされている。実行手段は、プログラム復号化手段が出力するプログラムを実行するようになされて

いる。

【0021】請求項5に記載の情報処理方法においては、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化し、その復号化された第1のキーを用いて、プログラムを暗号化したものを復号化し、その復号化されたプログラムを実行するようになされている。

【0022】請求項6に記載の記録媒体には、コンピュータに、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化させ、その復号化された第1のキーを用いて、プログラムを暗号化したものを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されている。

【0023】請求項8に記載の情報処理装置においては、暗号化手段が、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化するようになされている。

【0024】請求項9に記載の情報処理方法においては、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化するようになされている。

【0025】請求項10に記載の記録媒体には、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されている。

【0026】請求項11に記載の情報処理装置においては、キー復号化手段は、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化し、確認手段は、キー復号化手段により復号化された第1のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認するようになされている。供給手段は、確認手段により正当なものであることが確認された署名が付されているプログラムを、実行手段に供給するようになされている。

【0027】請求項12に記載の情報処理方法においては、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化し、その復号化された第1のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認し、署名が正当なものであることが確認された場合のみ、プログラムを実行するようになされている。

【0028】請求項13に記載の記録媒体には、コンピュータに、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化させ、その復号化された第1のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認させ、署名が正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラ

ムが記録されている。

【0029】請求項14に記載の情報処理装置においては、処理手段が、請求項11に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理するようになされている。

【0030】請求項15に記載の情報処理方法においては、請求項11に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理するようになされている。

【0031】請求項16に記載の記録媒体には、請求項11に記載の情報処理装置において署名が正当なものであると確認されるように処理されたプログラムが記録されている。

【0032】

【発明の実施の形態】以下に、本発明の実施の形態を説明するが、その前に、特許請求の範囲に記載の発明の各手段と以下の実施の形態との対応関係を明らかにするために、各手段の後の括弧内に、対応する実施の形態（但し、一例）を付加して、本発明の特徴を記述すると、次のようになる。

【0033】即ち、請求項1に記載の情報処理装置は、プログラムを実行するための処理を行う情報処理装置であって、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化する第1のキー復号化手段（例えば、図14に示す復号部84や、図22に示す復号部131など）と、第1のキー復号化手段により復号化された第1のキーを用いて、プログラムを暗号化したものを復号化するプログラム復号化手段（例えば、図14に示す復号部82や、図22に示す復号部132など）と、プログラム復号化手段が出力するプログラムを実行する実行手段（例えば、図14や図22に示すJava仮想マシン83など）とを備えることを特徴とする。

【0034】請求項2に記載の情報処理装置は、第2のキーが暗号化されている場合において、第2のキーを暗号化したものを、第3のキーを用いて復号化する第2のキー復号化手段（例えば、図22に示す復号部84など）をさらに備えることを特徴とする。

【0035】請求項8に記載の情報処理装置は、プログラムを処理する情報処理装置であって、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化する暗号化手段（例えば、図12に示すプログラムの処理ステップS7など）を備えることを特徴とする。

【0036】請求項11に記載の情報処理装置は、プログラムを実行するための処理を行う情報処理装置であって、プログラムを実行する実行手段（例えば、図17に示すJava仮想マシン83など）と、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化するキー復

号化手段（例えば、図17に示す復号部84など）と、キー復号化手段により復号化された第1のキーを用いて、プログラムに付されている署名が正当なものかどうかを確認する確認手段（例えば、図17に示す署名確認部103など）と、確認手段により正当なものであることが確認された署名が付されているプログラムを、実行手段に供給する供給手段（例えば、図17に示す仮想マシン入力制御部104など）とを備えることを特徴とする。

【0037】請求項14に記載の情報処理装置は、プログラムを処理する情報処理装置であって、請求項11に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムを処理する処理手段（例えば、図16に示すプログラムの処理ステップS22およびS23など）を備えることを特徴とする。

【0038】なお、勿論この記載は、各手段を上記したものに限定することを意味するものではない。

【0039】次に、本発明は、Javaのような仮想マシンの他、現実のマシンそのものについても適用可能であるが、ここでは、本発明を、Java仮想マシンに適用した場合を例に説明する。

【0040】なお、Javaについては、例えば、日経BP社発行の日経エレクトロニクスの1996.3.25（no.658）、同1996.6.17（no.664）などに、その詳細が記載されているので、ここでは、簡単に説明する。

【0041】Java仮想マシンは、抽象化された実行機械であり、その実態は、実際の計算機で実行されるプログラムである。そして、Java仮想マシンは、実際の計算機と同様に、プログラムカウンタや、スタックレジスタ、汎用レジスタ、スタックやヒープとしてのメモリ、その他の資源を有するが、これらの資源は、実際の計算機の資源にマッピングされている。

【0042】即ち、図1に示すように、いま実際の計算機1が、中央演算処理装置2、その中央演算処理装置2が内蔵するレジスタ3、メモリ4などの資源を有するとして、この計算機1に、Java仮想マシン11を実装すると、そのJava仮想マシン11の資源として、実際の計算機1の資源がマッピングされる。図1の実施の形態では、Java仮想マシン11は、資源として、レジスタ13や、メモリ14などを有しており、レジスタ13はレジスタ3に、メモリ14の200番地は、メモリ4の100番地に、それぞれマッピングされている。

【0043】実際の計算機1では、中央演算処理装置2に対する命令は、その資源に対する操作として実行されるが、Java仮想マシン11においても、その資源に対する操作として実行される命令が定義されている。このJava仮想マシン11に対する命令を記述するための言語がJava言語であり、Java仮想マシン11では、このJava言語で記述されたソースプログラム

を、Javaコンパイラで、Javaバイトコードにコンパイルしたものが、解釈、実行される。

【0044】即ち、図2に示すように、Java言語で記述されたソースプログラムであるJava言語プログラムは、Javaコンパイラ21でコンパイルされ、Javaバイトコードとされる。このJavaバイトコードが、Java仮想マシン11に入力され、Java仮想マシン11では、Javaバイトコードが、実際の計算機1（中央演算処理装置2）が解釈することのできる機械語コード（マシン語）に変換される。具体的には、例えば、図3に示すように、「move #125, レジスタ13」というJavaバイトコードで記述された「数字の125を、レジスタ13にセット」という命令（Javaバイトコード命令）が、Java仮想マシン11に入力された場合、Java仮想マシン11では、このJavaバイトコードが、「move #125, レジスタ3」という機械語コードで記述された命令（マシン語命令）に変換される。

【0045】そして、計算機1では、この機械語コードで記述された命令が実行されることにより、図4（A）に示すように、計算機1のレジスタ3に、数字の125がセットされる。

【0046】上述したように、計算機1のレジスタ3には、Java仮想マシン11のレジスタ13がマッピングされており、従って、計算機1のレジスタ3に、数字の125をセットすることは、Java仮想マシン11から見れば、図4（B）に示すように、そのレジスタ13に数字の125をセットすることになる。

【0047】以上のように、Java仮想マシン11へのJavaバイトコードによる命令は、計算機1用の機械語コードに変換され、Java仮想マシン11の資源にマッピングされている計算機1の資源に対する操作として実行される。この操作は、Java仮想マシン11から見れば、その資源（Java仮想マシン11の資源）に対する操作となり、Javaバイトコードによる命令が実行されたことになる。

【0048】従って、前述したように、Java仮想マシンを、実際の計算機（コンピュータ）に実装することで、その計算機が使用しているCPU（Central Processing Unit）やOSに無関係に、コンパイル済みのJavaプログラムを実行することができる。

【0049】ここで、Javaバイトコードを機械語コードに変換して実行する手法としては、例えば、Basic言語によるプログラムを実行する場合に採用されている、各命令を機械語コードに逐次翻訳して実行するインタープリタ形式と、命令を一括して機械語コードに翻訳して実行するJIT（Just In Time）コンパイラ形式などがある。

【0050】なお、Basic言語によるプログラムを実行する場合に採用されているインタープリタ形式は、

そのソースコードを翻訳する点で、中間コードであるJavaバイトコードを翻訳する場合と異なるが、ここでは、これらを特に区別しない（区別する必要もない）。

【0051】次に、図5は、本発明を適用したプログラム提供システム（システムとは、複数の装置が論理的に集合したものをいい、各構成の装置が同一筐体中にあるか否かは問わない）の一実施の形態の構成例を示している。

【0052】このプログラム提供システムにおいては、ソフトウェア開発者が、プログラム認証機関により認証されていないアプリケーションプログラムをユーザに配布した場合に、そのユーザ端末33上での、そのアプリケーションプログラムの実行を制限するようになされている。

【0053】即ち、ソフトウェア開発者は、例えば、プログラム認証機関、またはプログラム認証機関にプログラムの認証を依頼している者が開発したプログラム実行環境としてのJava仮想マシン上で動作するアプリケーションプログラムを開発し、その配布を希望する場合には、プログラム認証機関との間で、その旨のライセンス契約を結ぶ。ライセンス契約を結ぶと、プログラム認証機関からソフトウェア開発者に対しては、アプリケーションプログラムの認証としての、そのアプリケーションプログラムの暗号化、またはアプリケーションプログラムへの署名の付加に用いるキーが配布される。

【0054】具体的には、ここでは、アプリケーションプログラムに対する暗号化または署名の付加が、例えば、RSA方式（MITの3人の研究者により考案されたもので、RSAは、その3人の研究者の頭文字をとったものである）などに代表される公開鍵暗号化方式によって行われるものとする、その公開鍵暗号化方式による暗号化に用いられる秘密キーAと、その秘密キーAを用いた暗号化結果を復号化するのに用いる公開キーAとのセットが配布される。

【0055】ここで、ソフトウェア開発者が有するソフトウェア開発者サーバ31と、プログラム認証機関が有するプログラム認証機関サーバ32とは、例えば、インターネットや、公衆回線、CATV網、地上波、衛星回線、その他でなるネットワーク34を介して通信することができるようになされており、秘密キーAおよび公開キーAのセットの配布は、例えば、このネットワーク34を介して行われる。

【0056】なお、秘密キーAおよび公開キーAは、プログラム認証機関が用意して、ライセンス契約を結んだソフトウェア開発者に配布する他、ソフトウェア開発者が、自身で用意するようにしても良い。

【0057】ソフトウェア開発者は、秘密キーAおよび公開キーAの配布を受けた後、そのうちの公開キーAを、例えば、そのソフトウェア開発者サーバ31から、ネットワーク34を介して、プログラム認証機関サーバ

32に送信する。

【0058】プログラム認証機関サーバ32は、ソフトウェア開発者サーバ31から公開キーAを受信すると、それを暗号化し、その暗号化した公開キー（以下、適宜、暗号化公開キーという）Aを、ネットワーク34を介して、ソフトウェア開発者サーバ31に送信する。

【0059】ここで、プログラム認証機関サーバ32において、公開キーAの暗号化は、例えば、公開鍵暗号化方式により行われる。即ち、プログラム認証機関サーバ32では、暗号化に用いる秘密キーBと、その秘密キーBを用いた暗号化結果を復号化するのに用いる公開キーBとのセットが用意されており、公開キーAの暗号化は、その秘密キーBを用いて行われる。

【0060】一方、ソフトウェア開発者サーバ31では、プログラム認証機関サーバ32からの暗号化公開キーAが受信され、その後、秘密キーAを用いての、アプリケーションプログラムの暗号化、またはアプリケーションプログラムへの署名の付加が行われる。そして、暗号化されたアプリケーションプログラム、または署名の付加されたアプリケーションプログラムが、暗号化公開キーAと対応付けられて記憶される。

【0061】そして、ソフトウェア開発者サーバ31は、例えば、ネットワーク34を介して、ユーザ端末33からアプリケーションプログラムの要求があると、そのアプリケーションプログラム（上述したように、暗号化されたアプリケーションプログラム、または署名の付加されたアプリケーションプログラム）を、それに対応付けられた暗号化公開キーAとともに、ネットワーク34を介して、ユーザ端末33に送信する。

【0062】ユーザ端末33には、プログラム認証機関、またはプログラム認証機関にプログラムの認証を依頼している者が開発したプログラム実行環境としてのJava仮想マシンが実装されている。即ち、プログラム認証機関サーバ32には、そのプログラム認証機関が提供するJava仮想マシンを含む、Javaプログラムの実行環境としてのプログラム実行システム（コンピュータ（ユーザ端末33）を、プログラム実行システムとして動作させるためのプログラム）が記憶されており、ユーザが、ユーザ端末33を操作することにより、そのプログラム実行システムを、プログラム認証機関サーバ32に要求すると、プログラム認証機関サーバ32は、プログラム実行システムを、例えば、ネットワーク34を介して、ユーザ端末33に送信する。ユーザ端末33においては、このようにしてプログラム認証機関から提供されるプログラム実行システムが実装されている。

【0063】なお、プログラム認証機関サーバ32からユーザ端末33に対しては、プログラム実行システムとともに、公開キーBも送信されるようになされており、この公開キーBは、ユーザ端末33において記憶されるようになされている。

【0064】そして、プログラム実行システムが実装された、Java仮想マシンとしてのユーザ端末33では、ソフトウェア開発者サーバ31から受信したアプリケーションプログラムが、プログラム認証機関によって認証されているものであるときのみ、そのアプリケーションプログラムが、プログラム実行システム上において、正常に実行される。

【0065】即ち、ユーザ端末33では、ソフトウェア開発者サーバ31から受信した暗号化公開キーAが、プログラム認証機関サーバ32から受信した公開キーBを用いて、公開キーAに復号化される。さらに、ユーザ端末33では、その公開キーAを用いて、ソフトウェア開発者サーバ31から受信した、暗号化されたアプリケーションプログラムが復号化され、その復号化されたアプリケーションプログラムが、Java仮想マシン上で実行される。あるいは、また、ソフトウェア開発者サーバ31から受信したアプリケーションプログラムに付加された署名の正当性が、公開キーAを用いて確認され、署名の正当性が確認された場合のみ、アプリケーションプログラムが、Java仮想マシン上で実行される。

【0066】従って、公開キーBを用いて復号化されるキーが、プログラム認証機関から発行された、いわば正当な暗号化公開キーAではない場合、その復号化結果として、公開キーAを得ることはできない。このため、そのような復号化結果を用いて、アプリケーションプログラムを復号化しても、Java仮想マシン上で正常な処理が行われるようなアプリケーションプログラムを得られず、結局、Java仮想マシンは正常動作しない。

【0067】同様に、公開キーAでないキーを用いて署名を確認した場合には、その署名の正当性は否定されるため、やはり、Java仮想マシンでは、アプリケーションプログラムは実行されない。

【0068】以上のように、ソフトウェア開発者サーバ31から受信したアプリケーションプログラムが、プログラム認証機関によって認証されていない場合には、Java仮想マシンとしてのユーザ端末33では、そのアプリケーションプログラムは実行されない（実行されても、正常には実行されない）。

【0069】従って、結果として、プログラム実行環境としてのJava仮想マシンを開発、配布した者は、そのJava仮想マシンにおいて実行される、第三者により開発されたアプリケーションプログラムの、いわば勝手な配布を制限し、例えば、ライセンス契約を結んだソフトウェア開発者だけに、アプリケーションプログラムの配布を許可することができる。

【0070】さらに、上述の場合においては、ソフトウェア開発者は、公開キーAおよび秘密キーA、並びに暗号化公開キーAを得た後は、ユーザ端末33に実装されたJava仮想マシンで実行可能なアプリケーションプログラムの配布を自由に行うことができる。従って、新

たなアプリケーションプログラムを開発することに、そのアプリケーションプログラムを、プログラム認証機関サーバ32に送信して認証してもらう手間を省くことができる。

【0071】但し、プログラム認証機関側からすれば、ソフトウェア開発者による公開キーAおよび秘密キーA、並びに暗号化公開キーAの使用を、幾つかのアプリケーションプログラムに制限したい場合があるが、これは、例えば、プログラム認証機関が、公開キーAを暗号化するのに用いる秘密キーBを変更することによって対処可能である（この場合、ユーザには、その変更後の秘密キーに対する公開キーを配布する必要がある）。また、そのような変更を行わないまでも、公開キーAおよび秘密キーAとして、ソフトウェア開発者ごとにユニークなものを割り当てるようにすれば、暗号化公開キーAや、署名を参照することにより、アプリケーションプログラムを配布したソフトウェア開発者を特定することができる。従って、例えば、公開キーAおよび秘密キーA、並びに暗号化公開キーAの使用を、幾つかのアプリケーションプログラムに制限する旨のライセンス契約をしたのにも拘らず、その数を越えたアプリケーションプログラムの配布を行っているソフトウェア開発者を、容易に特定することができる。

【0072】なお、ソフトウェア開発者は、アプリケーションプログラムを、例えば、CD（Compact Disc）－ROM、磁気ディスク、その他でなる記録媒体35に記録して、郵送、店頭における販売、その他の手段で、ユーザに配布することも可能であるが、この場合においても、上述の場合と同様に、そのアプリケーションプログラムが、プログラム認証機関によって認証されていないときには、Java仮想マシンとしてのユーザ端末33では、そのアプリケーションプログラムは実行されない。

【0073】また、上述の場合においては、ソフトウェア開発者とプログラム認証機関との間で、ネットワーク34を介して、データ（ここでは、公開キーAおよび秘密キーA、並びに暗号化公開キーA）のやりとりを行うようにしたが、その他、データのやりとりは、そのデータを記録した記録媒体（図示せず）を郵送などすることによって行うことも可能である。

【0074】さらに、プログラム認証機関からユーザへのプログラム実行システムおよび公開キーBの提供も、例えば、CD－ROM、磁気ディスク、その他でなる記録媒体36に記録して、郵送、店頭における販売、その他の手段で行うことが可能である。

【0075】さらに、図5の実施の形態では、ソフトウェア開発者サーバ31、プログラム認証機関サーバ32、またはユーザ端末33を、それぞれ1つしか設けていないが、これらは、複数設けることが可能である。

【0076】次に、図6は、図5のソフトウェア開発者

サーバ31の構成例を示している。

【0077】CPU41は、補助記憶装置46に記憶（記録）されたオペレーティングシステムの制御の下、同じく補助記憶装置46に記憶されたプログラムを実行することで、各種の処理を行う。ROM（Read Only Memory）42は、例えば、IPL（Initial Program Loading）のプログラムなどを記憶している。RAM（Random Access Memory）43は、CPU41が実行するプログラムや、CPU1の動作上必要なデータを記憶する。入力部44は、例えば、キーボードやマウスなどで構成され、所定のデータやコマンドを入力するときなどに操作される。出力部45は、ディスプレイやプリンタなどで構成され、所定の情報を表示、印刷する。補助記憶装置46は、例えば、ハードディスクなどで構成され、オペレーティングシステム、その他のCPU41が実行するプログラムなどを記憶している。さらに、補助記憶装置46は、CPU41の処理結果その他の必要なデータなども記憶する。通信制御部47は、ネットワーク34を介して行われる通信を制御する。

【0078】次に、図7は、図5のプログラム認証機関サーバ32の構成例を、図8は、図5のユーザ端末33の構成例を、それぞれ示している。

【0079】プログラム認証機関サーバ32は、CPU51乃至通信制御部57で構成され、また、ユーザ端末33は、CPU61乃至通信制御部67で構成され、これらは、図6におけるCPU41乃至通信制御部47とそれぞれ同様に構成されるので、その説明は省略する。

【0080】次に、公開キーAやアプリケーションプログラムの暗号化／復号化の方法の1つとしての、上述した公開鍵暗号化方式による暗号化／復号化について、さらに説明する。

【0081】図9は、公開鍵暗号化方式による暗号化／復号化システムの構成例を示している。

【0082】暗号化器71には、暗号化対象である平文が入力される。そして、暗号化器71では、秘密キーと呼ばれる各個人に特有の暗号キーを用いて、平文が暗号化され、暗号文とされる。

【0083】一方、復号化器72には、暗号化器71で暗号化された暗号文が入力される。そして、復号化器72では、公開キーと呼ばれる広く一般に公開される復号キーを用いて、暗号文が復号化され、元の平文とされる。

【0084】ソフトウェア開発者サーバ31では、アプリケーションプログラムとしてのJavaバイトコードが、暗号化器71における場合と同様にして、秘密キーAを用いて暗号化される。また、プログラム認証機関サーバ32では、ソフトウェア開発者サーバ31からの公開キーAが、やはり暗号化器71における場合と同様にして、秘密キーBを用いて暗号化される。

【0085】一方、ユーザ端末33では、復号化器72

における場合と同様にして、暗号化公開キーAが、公開キーBを用いて復号化され、さらに、その復号化された公開キーAを用いて、暗号化されたアプリケーションプログラムが復号化される。

【0086】ここで、暗号化／復号化の手法は、公開鍵暗号化方式に限定されるものではなく、その他、例えば、DES (Data Encryption Standard) 方式 (IBM社により開発され、米国連邦政府の標準として実用化されたもの) などに代表される共通鍵暗号化方式その他の方式を採用することが可能である。なお、共通鍵暗号化方式では、当事者間以外に公開されない共通キーを用いて暗号化／復号化が行われる。公開鍵暗号化方式では、暗号化に用いられる秘密キーと復号化に用いられる公開キーとは異なるが (逆に、公開キーを暗号化に、秘密キーを復号化に、それぞれ用いてもかまわない)、共通鍵暗号化方式では、暗号化と復号化に、同一のキーである共通キーが用いられるので、その共通キーは、当事者間以外の者には秘密にしておく必要がある。

【0087】次に、アプリケーションプログラムに署名 (デジタルサイン) を付加する方法の1つとしての公開鍵暗号化方式を利用した署名付加方法について説明する。

【0088】図10は、公開鍵暗号化方式を利用した署名の作成／確認を行う暗号化／復号化システムの構成例を示している。

【0089】ダイジェスト作成器91には、暗号化対象である平文が入力される。そして、ダイジェスト作成器91では、入力された平文のダイジェストが、例えば、MD5やSHA-1などのアルゴリズムにしたがって作成される。

【0090】ここで、ダイジェストは、平文の機械的な凝縮文に相当し、入力としての平文が異なれば、そのダイジェストも異なるものが作成される。ダイジェストの作成は、平文を、例えば、ハッシュ関数を用いて変換することで行われる。

【0091】なお、データベースの検索を行うのに用いるキーワードがとり得る範囲の集合を、ある限られた数値範囲 (レコード番号や配列の添字などに対応する) に写像する方法は、ハッシング (hashing) と呼ばれるが、この写像を行う変換関数がハッシュ関数である。

【0092】ダイジェスト作成器91で作成されたダイジェストは、暗号化器92に供給される。暗号化器92では、例えば、図9の暗号化器71における場合と同様にして、ダイジェストが、秘密キーを用いて暗号化され、この暗号化されたダイジェストがデジタルサイン (署名) として出力される。そして、デジタルサインが、元の平文に付加され、署名付き平文として出力される。

【0093】一方、復号化器93には、署名付き平文を構成するデジタルサインが、また、ダイジェスト作成

器94には、その残りの平文が、それぞれ入力される。復号化器93では、例えば、図9の復号化器72における場合と同様にして、デジタルサインが、公開キーを用いて復号化され、ダイジェストとされる。このダイジェストは、署名確認器95に供給される。

【0094】ダイジェスト作成器94では、ダイジェスト作成器91における場合と同様にして、そこに入力される平文のダイジェストが作成され、署名確認器95に供給される。

【0095】署名確認器95では、署名 (デジタルサイン) の正当性が判断される (署名の確認が行われる)。即ち、署名確認器95では、復号化器93が出力するダイジェストが、ダイジェスト作成器94が出力するダイジェストと一致するかどうかを確認される。復号化器93が出力するダイジェストが、ダイジェスト作成器94が出力するダイジェストと一致しない場合、例えば、平文の改竄が行われたとして、あるいは、復号化器93で用いられた公開キーが正しくないとして、署名の正当性が否定される。

【0096】一方、復号化器93が出力するダイジェストが、ダイジェスト作成器94が出力するダイジェストと一致する場合、平文の改竄が行われておらず、かつ復号化器93で用いられた公開キーが正しいとして、署名の正当性が確認される。

【0097】署名確認器95には、署名付き平文を構成する平文も供給されるようになされており、ここでは、署名の正当性が確認されると、その平文が出力される。

【0098】ソフトウェア開発者サーバ32において、アプリケーションプログラムに、その署名が付される場合においては、上述したように、アプリケーションプログラムのダイジェストが作成され、そのダイジェストが秘密キーAを用いて暗号化されることにより、署名 (デジタルサイン) が作成される。一方、ユーザ端末33においては、その署名が公開キーAを用いてダイジェストに復号化されるとともに、アプリケーションプログラムからダイジェストが作成され、その2つのダイジェストが一致するかどうかで、署名の正当性が確認される。

【0099】署名は、その署名をした者 (ここでは、ソフトウェア開発者) を証明するものであるから、署名が付加されたアプリケーションプログラムについては、その署名から、その出所を、容易に特定することができる。従って、例えば、アプリケーションプログラムにバグなどがある場合に、そのようなバグのあるアプリケーションプログラムを配布したソフトウェア開発者を、容易に見つけ出すことができる。

【0100】さらに、署名が付加されたアプリケーションプログラムが改竄されていたり、いわゆるコンピュータウイルスなどに侵されている場合においては、そのアプリケーションプログラムから作成されるダイジェストと、署名を復号化して得られるダイジェストとは一致せ

ず、署名の正当性が否定されるから、そのような改竄やコンピュータウイルスに侵されたアプリケーションプログラムが、ユーザ端末33において実行されることを防止することが可能となる。

【0101】また、署名の作成に用いられる秘密キーAは、第三者（ここでは、ソフトウェア開発者およびプログラム認証機関以外の者）に秘密にされるものであるから、仮に、第三者が、アプリケーションプログラムの解読などを行っても、署名の作成方法まで知ることは困難である。

【0102】なお、署名の作成／確認方法は、上述した公開鍵暗号化方式を利用したものに限定されるものではない。

【0103】次に、アプリケーションプログラムの認証が、秘密キーAを用いた暗号化により行われる場合のソフトウェア開発者サーバ31、プログラム認証機関サーバ32、およびユーザ端末33の処理について説明する。

【0104】まず、図11および図12のフローチャートを参照して、ソフトウェア開発者サーバ31の処理について説明する。

【0105】ソフトウェア開発者は、プログラム認証機関とライセンス契約を結び、公開キーAおよび秘密キーAを、例えば、プログラム認証機関から発行してもらう等して取得する。そして、ソフトウェア開発者サーバ31では、まず最初に、図11のステップS1において、通信制御部47が、公開キーAをプログラム認証機関に認証してもらうために、ネットワーク34を介して、プログラム認証機関サーバ32に送信し、ステップS2に進む。ステップS2では、CPU41において、認証された公開キーAとしての、その公開キーAを暗号化した暗号化公開キーAが、プログラム認証機関サーバ32から送信されてきたかどうか判定され、まだ、送信されてきていないと判定された場合、ステップS2に戻る。

【0106】また、ステップS2において、暗号化公開キーAが送信されてきたと判定された場合、ステップS3に進み、通信制御部47において、その暗号化公開キーAが受信され、ステップS4に進む。ステップS4では、通信制御部47で受信された暗号化公開キーが、補助記憶装置46に転送されて記憶され、処理を終了する。

【0107】その後、ソフトウェア開発者が、Java仮想マシン上で実行されるアプリケーションプログラムを開発し、そのアプリケーションプログラムが、例えば、補助記憶装置46に記録（記憶）されると、ソフトウェア開発者サーバ31では、図12のステップS6において、CPU41が、補助記憶装置46に記憶されたアプリケーションプログラムを、Javaコンパイラのプログラムにしたがってコンパイルし、Javaバイトコードとする。このJavaバイトコードは、再び、補

助記憶装置46に供給されて記憶される。

【0108】そして、ステップS7に進み、CPU41において、ステップS6のコンパイルの結果得られたJavaバイトコードが、例えば、図9で説明したように、秘密キーAを用いて暗号化され、暗号化バイトコードとされる。この暗号化バイトコードは、ステップS8において、暗号化公開キーAと対応付けられて、補助記憶装置46に記憶され、処理を終了する。

【0109】次に、図13のフローチャートを参照して、プログラム認証機関サーバ32の処理について説明する。

【0110】プログラム認証機関は、例えば、プログラム実行環境としてのJava仮想マシンを開発した者、あるいは、その依頼を受けた者の機関で、そのプログラム認証機関サーバ32では、例えば、ライセンス契約を結んだ者のアプリケーションプログラムの認証として、公開キーAの認証を行う認証処理を行うようになされている。

【0111】即ち、プログラム認証機関サーバ32のCPU51では、まず最初に、ステップS11において、例えば、ソフトウェア開発者サーバ31などから、認証対象の公開キーAが、ネットワーク34を介して送信されてきたかどうか判定され、送信されてきていないと判定された場合、ステップS11に戻る。また、ステップS11において、公開キーAが送信されてきたと判定された場合、ステップS12に進み、その公開キーAが、例えば、ライセンス契約を結んだ、いわば正規のソフトウェア開発者からのものであるかどうか、CPU51によって判定される。

【0112】即ち、プログラム認証機関は、Java仮想マシン上で実行されるアプリケーションプログラムの開発、配布等を許可することについてのライセンス契約を、ソフトウェア開発者との間で結ぶと、そのソフトウェア開発者に対して、例えば、IDおよびパスワードを発行するようになされている。そして、このようなライセンス契約を結んだ正規のソフトウェア開発者からは、認証対象の公開キーAとともに、ライセンス契約時に発行されたIDおよびパスワードが送信されるようになされており、ステップS12では、このIDおよびパスワードに基づいて、公開キーAが、正規のソフトウェア開発者からのものであるかどうか判定される。

【0113】ステップS12において、公開キーAが、正規のソフトウェア開発者からのものでないと判定された場合、即ち、ライセンス契約を結んでいないソフトウェア開発者から公開キーAが送信されてきた場合、ステップS13に進み、通信制御部57において、そのソフトウェア開発者に対して、ライセンス契約を結ばなければ、公開キーAを認証することができない旨のメッセージが送信され、処理を終了する。

【0114】一方、ステップS12において、公開キー

Aが、正規のソフトウェア開発者からのものであると判定された場合、ステップS14に進み、CPU51において、その公開キーAが暗号化され、これにより暗号化公開キーAとされることで、その認証が行われる。

【0115】そして、ステップS15に進み、通信制御部57において、公開キーAの認証結果としての暗号化公開キーAが、その公開キーAを送信してきたソフトウェア開発者、即ち、ここでは、例えば、ソフトウェア開発者サーバ31に、ネットワーク34を介して送信され、処理を終了する。

【0116】次に、図14は、ユーザ端末33においてアプリケーションプログラムを実行するプログラム実行環境としてのプログラム実行システムの機能的構成例を示している。

【0117】入力部81は、暗号化バイトコード（暗号化されたJavaバイトコード）および暗号化公開キーA（暗号化された公開キーA）、並びに公開キーBを受け付け、暗号化バイトコードを復号部82に、暗号化公開キーAおよび公開キーBを復号部84に、それぞれ供給するようになされている。復号部82は、図11の復号化器72としての処理を行うもので、入力部81からの暗号化バイトコードを、復号部84から出力される公開キーAを用いて復号化し、元のJavaバイトコードとするようになされている。復号部82で得られたJavaバイトコードは、Java仮想マシン83に供給されるようになされており、Java仮想マシン83は、復号部82からのJavaバイトコードにしたがった処理を実行するようになされている。復号部84も、復号部82と同様に、図11の復号化器72としての処理を行うもので、入力部81からの暗号化公開キーAを、同じく入力部81からの公開キーBを用いて、公開キーAに復号化し、復号部82に供給するようになされている。

【0118】以上のように構成されるプログラム実行システムでは、まず最初に、入力部81において、暗号化バイトコードおよび暗号化公開キーA、並びに公開キーBが取得される。即ち、例えば、あらかじめ、ソフトウェア開発者サーバ31から、ネットワーク34を介して、暗号化バイトコードおよび暗号化公開キーAを受信し、それらが、ファイルとして、補助記憶装置67に記憶されている場合や、暗号化バイトコードおよび暗号化公開キーAがファイルとして記録された記録媒体35がユーザ端末33にセットされる場合などには、入力部81は、そのファイルをオープンし、暗号化バイトコードおよび暗号化公開キーAを読み出す。

【0119】また、例えば、ソフトウェア開発者サーバ31が、インターネットとしてのネットワーク34に接続されており、そのようなソフトウェア開発者サーバ31において、暗号化バイトコードおよび暗号化公開キーAが、URL (Uniform Resource Locator) と対応付け

られている場合に、ユーザが、入力部64を操作して、そのURLを指定したときには、入力部81は、ソフトウェア開発者サーバ31からネットワーク34を介して送信されてくる暗号化バイトコードおよび暗号化公開キーAを受信する。

【0120】さらに、例えば、ソフトウェア開発者サーバ31が、地上波や衛星回線としてのネットワーク34を介して、暗号化バイトコードおよび暗号化公開キーAをデジタル放送などしている場合には、入力部81は、その放送されてくる暗号化バイトコードおよび暗号化公開キーAを受信する。

【0121】入力部81は、同様にして、プログラム認証機関が発行する公開キーBも取得する。

【0122】入力部81は、以上のようにして取得した暗号化バイトコードおよび暗号化公開キーA、並びに公開キーBのうち、暗号化バイトコードを復号部82に、暗号化公開キーAおよび公開キーBを復号部84に、それぞれ供給する。

【0123】復号部84では、入力部81からの暗号化公開キーAが、同じく入力部81からの公開キーBを用いて、公開キーAに復号化され、復号部82に供給される。復号部82では、復号部84からの公開キーAを用いて、入力部81からの暗号化バイトコードが復号化され、その復号結果としてのJavaバイトコードが、Java仮想マシン83に供給される。Java仮想マシン83では、復号部82からのJavaバイトコードが解釈、実行される。

【0124】以上のように、復号部84において、プログラム認証機関サーバ32での暗号化に用いられた秘密キーBに対応する公開キーB（秘密キーBと対になる公開キーB）を用いて、暗号化公開キーAが、公開キーAに復号化される。さらに、復号部82において、ソフトウェア開発者サーバ31での暗号化に用いられた秘密キーAに対応する公開キーである、復号部84で復号化された公開キーAを用いて、暗号化バイトコードの復号化が行われ、その復号結果としてのJavaバイトコードが、Java仮想マシン83に入力される。

【0125】従って、公開キーBを用いて復号化されるキーが、プログラム認証機関から発行された、いわば正当な暗号化公開キーAではない場合、即ち、例えば、暗号化されていない公開キーAや、プログラム認証機関サーバ32における暗号化アルゴリズムと異なるアルゴリズムで暗号化された暗号化キーA、あるいはそれと同一のアルゴリズムで暗号化されていたとしても、本来使用すべき秘密キーBを用いずに暗号化された公開キーAなどが入力された場合には、その復号化結果として、正当な公開キーAを得ることはできない（正当な公開キーAが、復号部84から偶然に出力されることは有り得ないことではないが、ほとんどないに等しい）。このため、そのような復号化結果を用いて、暗号化バイトコードを

復号化しても、Java仮想マシン83上で正常な処理が行われるようなアプリケーションプログラムを得られず、結局、Java仮想マシン83は正常動作しない。よって、結果として、Java仮想マシン83上で動作するJavaバイトコードであって、プログラム認証機関で認証されていないものの、Java仮想マシン83が実装されたユーザ端末33を有するユーザへの配布を制限することができる。

【0126】即ち、Java仮想マシン83上で動作するJavaバイトコードの、そのJava仮想マシン83が実装されたユーザ端末33を有するユーザへの配布は、プログラム認証機関とライセンス契約を結んだソフトウェア開発者にのみ許可することができ、Java仮想マシン83の開発者等は、Java仮想マシン83上で動作するJavaバイトコードの配布を希望するソフトウェア開発者から、いわば、Java仮想マシン83を利用するためのライセンス料を得ることが可能となる。

【0127】なお、Java仮想マシン83に対するJavaバイトコードの入力は、復号部82からのみ行うことができるようにしておく必要があり、また、復号部82への公開キーAの入力も復号部84からのみ行うことができるようにしておくのが望ましい。

【0128】ここで、図14における復号部82および84では、何らかの入力があると、その入力に対して復号化処理が施され、その処理結果が出力される。従って、公開キーBでないものや、秘密キーAで暗号化されていないJavaバイトコード、秘密キーBで暗号化されていない公開キーAが入力され、そのようなものから得られる復号化結果が、Java仮想マシン83に供給されると、Java仮想マシン83は、通常暴走する。そこで、復号部82の出力が、正当（正常）なJavaバイトコードかどうかを確認し、正当なJavaバイトコードである場合にのみ、そのJavaバイトコードを、Java仮想マシン83に解釈、実行させるようにすることが可能である。即ち、例えば、Javaバイトコードには、その先頭に、マジック（magic）と呼ばれる32ビットのデータが配置されており、これが本来の値（16進数で、CAFEBABE）である場合に、復号部82の出力が正当なJavaバイトコードであるとして、Java仮想マシン83に解釈、実行させるようにすることができる。この場合、Java仮想マシン83の暴走を防止することができる。

【0129】ところで、図14の復号部82および84における復号化アルゴリズムや、公開キーA、Bが、ライセンス契約をしている者以外の者に知られても、Java仮想マシン83上におけるアプリケーションプログラムの実行の制限の観点からは、暗号化アルゴリズムやその暗号化に用いる秘密キーA、Bさえ知らなければ、特に問題はない。即ち、暗号文の復号化方法が知ら

れても、正常に実行可能なJavaバイトコードをJava仮想マシン83に供給するために復号部82に与えるべき暗号化バイトコードの作成方法や、復号部84に与えるべき暗号化公開キーAの作成方法を知らなければ、仮想マシン83上におけるアプリケーションプログラムの実行を制限することができる。

【0130】しかしながら、暗号文の復号化方法が知られた場合、暗号化バイトコードから、元のJavaバイトコードを得ることができる。Javaバイトコードは、前述したように、それを逆コンパイルすることにより、その内容を比較的容易に理解することができるため、リバースエンジニアリングを簡単にすることができる。

【0131】そこで、このようなリバースエンジニアリングを防止すべく、暗号文の復号化方法は秘密にすることができる。即ち、例えば、暗号化バイトコードの復号化に用いる公開キーAを暗号化した暗号化公開キーAを復号化するための公開キーBは、一般に公開されるものであるが、これを秘密にすることができる。

【0132】図15は、公開キーBを秘密にするようにしたプログラム実行システムの構成例を示している。なお、図中、図14における場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【0133】この実施の形態においては、例えば、Java仮想マシン83を含むプログラム実行システムを構成するプログラムの中の1カ所に、または複数箇所に分散して、公開キーBが配置されており、復号部84では、その公開キーBを用いて、暗号文の復号化が行われる。従って、この場合、公開キーBは、プログラム実行システムの外部に漏れることがなく、その結果、暗号文が、不正に復号され、リバースエンジニアリングされることを防止すること（リバースエンジニアリングされる確率を低減すること）が可能となる。

【0134】次に、アプリケーションプログラムの認証が、秘密キーAを用いて作成された署名が付加されることにより行われる場合のソフトウェア開発者サーバ31およびユーザ端末33の処理について説明する。なお、この場合、プログラム認証機関サーバ32では、上述した、アプリケーションプログラムが暗号化される場合と同様の処理が行われるので、その説明は省略する。

【0135】まず、図16のフローチャートを参照して、ソフトウェア開発者サーバ31の処理について説明する。なお、ソフトウェア開発者サーバ31では、既に、図11のフローチャートで説明した処理が行われ、これにより、プログラム認証機関から暗号化公開キーAを取得しているものとする。また、ソフトウェア開発者においては、Java仮想マシン上で実行されるアプリケーションプログラムが開発され、補助記憶装置46に記憶されているものとする。

【0136】この場合も、ソフトウェア開発者サーバ31では、図12のステップS6における場合と同様に、ステップS21において、CPU41が、補助記憶装置46に記憶されたアプリケーションプログラムを、Javaコンパイラのプログラムにしたがってコンパイルし、Javaバイトコードとする。このJavaバイトコードは、再び、補助記憶装置46に供給されて記憶される。

【0137】そして、ステップS22以下に順次進み、ステップS21のコンパイルの結果得られたJavaバイトコードに対して、それが正当なものであることを証明する署名（デジタルサイン）が付される。

【0138】即ち、ステップS22では、CPU51において、Javaバイトコードのダイジェストが、例えば、図10のダイジェスト作成器91における場合と同様にして作成され、ステップS23に進む。ステップS23では、CPU51において、ステップS22で作成されたダイジェストが、例えば、図10の暗号化器92における場合と同様にして、秘密キーAを用いて暗号化されることにより、デジタルサイン（署名）が作成される。そして、ステップS24に進み、そのデジタルサインが、Javaバイトコードに付される（このようにデジタルサイン（署名）が付されたJavaバイトコードを、以下、適宜、署名付きバイトコードという）。さらに、ステップS24では、署名付きバイトコードが、暗号化公開キーAと対応付けられて、補助記憶装置46に記憶され、処理を終了する。

【0139】次に、図17は、ユーザ端末33においてアプリケーションプログラムの正当性を確認し、正当なもののみを実行するプログラム実行環境としてのプログラム実行システムの機能的構成例を示している。なお、図中、図14における場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【0140】入力部101は、基本的には、図12における入力部81と同様に、そこへの入力を受け付けるようになされている。但し、入力部101には、暗号化バイトコードの代わりに、署名付きバイトコード（署名（デジタルサイン）が付されたJavaバイトコード）が入力されるようになされており、そこでは、署名とJavaバイトコードとに分離されて出力されるようになされている。署名は、署名確認部103に、Javaバイトコードは、メッセージダイジェストシステム102および仮想マシン入力制御部104に、それぞれ供給されるようになされている。

【0141】メッセージダイジェストシステム102は、図10のダイジェスト作成器94と同様の処理を行うもので、入力部101からのJavaバイトコードからダイジェストを作成し、署名確認部103に供給するようになされている。署名確認部103は、図10の復

号化器93および署名確認器95に相当するもので、入力部101からの署名の正当性を確認するようになされている。

【0142】即ち、署名確認部103には、入力部101から署名が、メッセージダイジェストシステム102からダイジェストが、それぞれ供給される他、復号部84から、公開キーAが供給されるようになされている。そして、署名確認部103は、その公開キーAを用いて、署名を復号化することにより、ダイジェストとし、そのダイジェストと、メッセージダイジェストシステム102からのダイジェストとを比較することで、署名の正当性を確認するようになされている。さらに、署名確認部103は、その確認結果に対応して、仮想マシン入力制御部104を制御するようになされている。

【0143】仮想マシン入力制御部104は、署名確認部103の制御にしたがって、入力部101からのJavaバイトコードの、Java仮想マシン83への供給を制御するようになされている。

【0144】以上のように構成されるプログラム実行システムでは、まず最初に、入力部101において、図14の入力部81における場合と同様にして、署名付きバイトコードおよび暗号化公開キーA、並びに公開キーBが取得される。そして、入力部101は、署名付きバイトコードを、署名とJavaバイトコードとに分離し、署名を、署名確認部103に、Javaバイトコードを、メッセージダイジェストシステム102および仮想マシン入力制御部104に、それぞれ供給する。さらに、入力部101は、暗号化公開キーAと公開キーBを復号部84に供給する。復号部84では、図14で説明したように、暗号化公開キーAが、公開キーBを用いて、公開キーAに復号化され、署名確認部103に供給される。

【0145】一方、メッセージダイジェストシステム102では、入力部101からのJavaバイトコードからダイジェストが作成され、署名確認部103に供給される。署名確認部103では、復号部84からの公開キーAを用いて、入力部101からの署名が復号化されてダイジェストとされる。さらに、署名確認部103では、その復号化したダイジェストが、メッセージダイジェストシステム102からのダイジェストと比較され、それが一致するかどうかで、入力部101からの署名の正当性が確認される。

【0146】署名の正当性が確認された場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム102からのダイジェストと一致する場合、署名確認部103は、入力部101からのJavaバイトコードを、Java仮想マシン83に出力するようにな、仮想マシン入力制御部104を制御する。この場合、仮想マシン入力制御部104は、署名確認部103の制御にしたがい、入力部101からのJavaバイト

コードを、Java仮想マシン83に供給する。

【0147】従って、この場合、Java仮想マシン83では、入力部101から仮想マシン入力制御部104を介して供給されるJavaバイトコードが解釈、実行される。

【0148】一方、署名の正当性が確認されなかった場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム102からのダイジェストと一致しない場合、署名確認部103は、入力部101からのJavaバイトコードを、Java仮想マシン83に出力しないように、仮想マシン入力制御部104を制御する。

【0149】この場合、仮想マシン入力制御部104からJava仮想マシン83に対しては、入力部101からのJavaバイトコードは出力されず、従って、Java仮想マシン83では、特に処理は行われない。

【0150】以上から、アプリケーションプログラムの認証として署名を付す場合も、Java仮想マシン83上で動作するJavaバイトコードであって、プログラム認証機関で認証されていないものの、Java仮想マシン83が実装されたユーザ端末33を有するユーザへの配布を制限することができる。即ち、Java仮想マシン83上で動作するJavaバイトコードの、そのJava仮想マシン83が実装されたユーザ端末33を有するユーザへの配布は、プログラム認証機関とライセンス契約を結んだソフトウェア開発者にのみ許可することができ、Java仮想マシン83の開発者は、Java仮想マシン83上で動作するJavaバイトコードの配布を希望するソフトウェア開発者から、Java仮想マシン83を使用するためのライセンス料を得ることが可能となる。

【0151】また、署名を付す場合においては、上述したように、その署名を付したJavaバイトコードを改竄したもの、仮想マシン83上での実行を制限することなどができる。

【0152】なお、図17の実施の形態では、Java仮想マシン83に対するJavaバイトコードの入力は、仮想マシン入力制御部104からのみ行うことができるようにしておく必要がある。また、署名確認部103への公開キーAの入力は、復号部84のみから行うことができるようにするのが望ましい。

【0153】ここで、Javaバイトコードに署名を付す場合においては、Javaバイトコードを暗号化する場合と異なり、Javaバイトコードそのものが存在する。従って、プログラム実行システムが署名の正当性を確認しないもの（例えば、入力部101の出力であるJavaバイトコードが、Java仮想マシン83に直接入力されるような構成のもの）であれば、そのプログラム実行システムでは、何の制限もなく、Javaバイトコードを解釈して実行することができる。

【0154】即ち、逆にいえば、Javaバイトコードに署名を付す場合において、その実行を制限したいJava仮想マシンの開発者や販売者、さらには、ユーザ端末33にJava仮想マシンを実装して販売する者などは、そのプログラム実行システムを、図17に示したように構成すれば良いし、その実行の制限を特に希望しない者は、プログラム実行システムを署名の正当性を確認しないような構成とすれば良い。

【0155】次に、図5の実施の形態では、プログラム認証機関からユーザに対して、ソフトウェアであるプログラム実行システム（ユーザ端末33をプログラム実行システムとして機能させるためのプログラム）を、そのままユーザ端末33に実装することができる形で提供するようにしたが、このプログラム実行システムも、ソフトウェア開発者が提供するアプリケーションプログラムと同様に、暗号化したり、また署名を付して提供するようにすることができる。

【0156】例えば、プログラム認証機関からユーザに対して、プログラム実行システムに署名を付加して提供する場合においては、プログラム認証機関サーバ32においては、例えば、図18のフローチャートにしたがった処理が行われる。

【0157】即ち、この場合、プログラム認証機関サーバ32では、まず最初に、ステップS31において、プログラム実行システムとしてのプログラムが、ユーザ端末33のCPU61が実行可能なコード（以下、適宜、実行コードという）にコンパイルされる。このコンパイルは、CPU51によって行われる。そして、CPU51は、ステップS32において、ステップS31のコンパイルの結果得られた実行コードから、例えば、図10で説明したようにしてダイジェストを作成し、ステップS33に進む。ステップS33では、CPU51において、ステップS32で作成されたダイジェストが、例えば、公開キーAの暗号化に用いられる秘密キーBを用いて暗号化され、これにより署名（デジタルサイン）が作成される。そして、ステップS34に進み、そのデジタルサインが、実行コードに付され（このようにデジタルサイン（署名）が付された実行コードを、以下、適宜、署名付き実行コードという）、その署名付きバイトコードが、署名の作成に用いられた秘密キーBと対になる公開キーBと対応付けられて、補助記憶装置56に記憶され、処理を終了する。

【0158】次に、図19は、ユーザ端末33において、上述したような署名が付加されたプログラム実行システム（署名付き実行コード）を実装するためのローダ（プログラム実行システムの実装装置）の機能的構成例を示している。

【0159】この実施の形態では、ローダは、入力部111、メッセージダイジェストシステム112、署名確認部113、および実行システム114で構成され、入

力部111、メッセージダイジェストシステム112、または署名確認部113は、図17の入力部101、メッセージダイジェストシステム102、または署名確認部103とそれぞれ同様に構成されている。実行システム114は、ユーザ端末33のCPU61その他の実行コードを解釈して実行する部分に相当する。

【0160】以上のように構成されるローダでは、入力部111において、署名付き実行コードおよび公開キーBが取得される。そして、入力部111は、署名付き実行コードを、署名と実行コードとに分離し、署名を署名確認部113に、実行コードをメッセージダイジェストシステム112および実行システム114に供給する。さらに、入力部111は、公開キーBを署名確認部113に供給する。

【0161】メッセージダイジェストシステム112では、入力部111からの実行コードからダイジェストが作成され、署名確認部113に供給される。署名確認部113では、入力部111からの公開キーBを用いて、入力部111からの署名が復号化されてダイジェストとされる。さらに、署名確認部113では、その復号化したダイジェストが、メッセージダイジェストシステム112からのダイジェストと比較され、それが一致するかどうかで、入力部111からの署名の正当性が確認される。

【0162】署名の正当性が確認された場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム112からのダイジェストと一致する場合、署名確認部113は、入力部111からの実行コードを解釈して実行するように、実行システム114を制御する。この場合、実行システム114では、署名確認部113の制御にしたがい、入力部111からの実行コードが解釈、実行される。

【0163】一方、署名の正当性が確認されなかった場合、即ち、署名を復号化したダイジェストが、メッセージダイジェストシステム112からのダイジェストと一致しない場合、署名確認部113は、入力部111からの実行コードを無視するように、実行システム114を制御する。この場合、実行システム114では、署名確認部113の制御にしたがい、入力部111からの実行コードが無視され、従って、特に処理は行われない。

【0164】以上のように、プログラム実行システムに署名を付加する場合においては、その改竄などを防止することが可能となる。

【0165】なお、上述したように、プログラム実行システムは暗号化することも可能であり、この場合、プログラム実行システムのリバースエンジニアリングを防止することが可能となる。

【0166】プログラム実行システムに付加された署名の正当性が確認された場合、上述したように、実行システム114において入力部111からの実行コードが解

釈、実行され、これにより、ユーザ端末33には、例えば、図20に示すように、図17における場合と同様のプログラム実行システムが実装される。但し、この場合においては、復号部84への公開キーAの供給は、入力部101から行われるのではなく、図19のローダにおける入力部111から行われる。

【0167】ところで、アプリケーションプログラムや、プログラム実行システムのプログラムなどのように、比較的情報量の多いものを、例えば、RSA方式などの公開鍵暗号化方式で暗号化し、また、その復号化を行う場合においては、その処理に時間を要する。これに対して、DES方式などの共通鍵暗号化方式による暗号化／復号化は、情報量が多いものでも、比較的短時間に行うことができる。一方、RSA方式では、暗号化に用いる秘密キーと、その復号化に用いる公開キーとが異なるので、公開キーは、上述したように一般に公開しても問題はないが、DES方式では、暗号化と復号化とに同一の共通キーを用いるため、共通キーは、当事者以外の者に知られないように、厳重に管理する必要がある。

【0168】そこで、暗号化／復号化の処理を短時間で行うとともに、キーの管理を容易にする手法として、例えば、アプリケーションなどの比較的情報量の多いものの暗号化には、DES方式を採用し、そのDES方式の暗号化に用いる共通キーを、RSA方式で暗号化するものがある（このようにDES方式とRSA方式とを組み合わせた暗号化の手法を、以下、適宜、組合せ方式という）。

【0169】即ち、図21は、アプリケーションプログラムについて、組合せ方式による暗号化を行う場合のソフトウェア開発者サーバ31の機能的構成例を示している。

【0170】アプリケーションプログラムをJavaコンパイラでコンパイルして得られるJavaバイトコードは、暗号化器121に入力される。暗号化器121には、Javaバイトコードの他に、共通キーも入力されるようになされており、そこでは、Javaバイトコードが、共通キーを用いて、例えば、DES方式により暗号化され、これにより、暗号化バイトコード（暗号文）とされる。

【0171】暗号化器121に入力される共通キーは、暗号化器122にも入力されるようになされている。暗号化器121は、例えば、RSA方式による暗号化を行うもので、そこでは、共通キーが、秘密キーAを用いて暗号化される。

【0172】この場合、ソフトウェア開発者からユーザに対しては、暗号化バイトコード、秘密キーAで暗号化された共通キー（以下、適宜、暗号化共通キーという）、およびプログラム認証機関から取得した暗号化公開キーAが1セットとして配布される。

【0173】次に、図22は、アプリケーションプログ

ラムの暗号化が組合せ方式で行われる場合のユーザ端末33に実装されるプログラム実行システムの機能的構成例を示している。なお、図中、図14における場合と対応する部分については、同一の符号を付してあり、以下では、その説明は、適宜省略する。

【0174】復号部131は、例えばRSA方式により、暗号化共通キーを復号化するようになされており、その復号化結果としての共通キーを、復号部132に供給するようになされている。復号部132は、例えばDES方式により、暗号化バイトコードを復号化するようになされている。

【0175】以上のように構成されるプログラム実行システムでは、入力部81において、暗号化バイトコード（暗号化されたJavaバイトコード）、暗号化共通キー（暗号化された共通キー）、および暗号化公開キーA（暗号化された公開キーA）、並びに公開キーBが取得される。そして、入力部81は、暗号化公開キーAおよび公開キーBを復号部84に、暗号化共通キーを復号部131に、暗号化バイトコードを復号部132に、それぞれ出力する。

【0176】復号部84では、上述したように、暗号化公開キーAが、公開キーBを用いて、公開キーAに復号化される。そして、この公開キーAは、復号部131に供給される。復号部131では、入力部81からの暗号化共通キーが、復号部84からの公開キーAを用いて、共通キーに復号化され、復号部132に供給される。復号部132では、入力部81からの暗号化バイトコードが、復号部131からの共通キーを用いて、Javaバイトコードに復号化され、Java仮想マシン83に供給される。

【0177】以上のような組合せ方式によれば、キーの管理の簡単化および暗号化／復号化処理の高速化を図ることが可能となる。

【0178】また、組合せ方式は、アプリケーションプログラムを暗号化する場合の他、例えば、プログラム実行システムのプログラム（実行コード）を暗号化する場合などにも適用可能である。

【0179】なお、本発明は、上述したインタープリタ形式およびJITコンパイラ形式のいずれのJava仮想マシンにも適用可能であるし、また、Java仮想マシン以外の仮想マシン、さらには、例えば、C言語や、C++言語などの処理系のようにプログラム実行システムへの入力が機械語コードの場合や、Basic言語の処理系のようにプログラム実行システムへの入力がソースコードの場合などについても適用可能である。

【0180】また、図14や図17などの実施の形態においては、プログラム実行システムを1つしか設けていないが、ユーザ端末33には、このプログラム実行システムを複数設けることも可能である。例えば、入力部81や101を複数設けた場合においては、複数経路から

の暗号文や署名付きバイトコードの入力が可能となる。さらに、例えば、復号部82や84を複数設けた場合においては、複数の復号アルゴリズムにしたがって暗号文を復号することが可能となる。また、例えば、Java仮想マシン83を複数設けた場合には、複数のJavaバイトコード形式をサポートすることが可能となる。さらに、例えば、メッセージダイジェストシステム102および署名確認部103を複数設けた場合には、複数の手法それぞれで付された複数の署名の確認を行うことが可能となる。

【0181】また、本実施の形態では、1のキーを用いて暗号化や署名の作成を行うようにしたが、暗号化や署名の作成は、複数のキーを用いて行うことも可能である。即ち、例えば、暗号化は、複数のキーを順次用いて行うことができ、また、署名は、複数のキーに対応する数だけ作成するようにすることなどができる。

【0182】なお、本実施の形態におけるJavaバイトコードとは、例えば、Java Application, Java Applet, Java Beans, Java Class Libraryその他の多数の形態のJavaバイトコードをすべて含むものである。

【0183】

【発明の効果】請求項1に記載の情報処理装置および請求項5に記載の情報処理方法によれば、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものが、第2のキーを用いて復号化され、その復号化された第1のキーを用いて、プログラムを暗号化したものが復号化される。そして、その復号化されたプログラムが実行される。また、請求項6に記載の記録媒体には、コンピュータに、プログラムを暗号化したものを復号化するのに必要な第1のキーを暗号化したものを、第2のキーを用いて復号化させ、その復号化された第1のキーを用いて、プログラムを暗号化したものを復号化させ、その復号化されたプログラムを実行させるためのプログラムが記録されている。従って、第1のキーが復号化され、さらに、プログラムが復号化された場合のみ、そのプログラムが実行されるようにすることが可能となる。

【0184】請求項8に記載の情報処理装置および請求項9に記載の情報処理方法によれば、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムが暗号化される。また、請求項10に記載の記録媒体には、請求項1に記載の情報処理装置で実行可能なコードに復号化される暗号文に、プログラムを暗号化したものが記録されている。従って、請求項1に記載の情報処理装置で実行可能な、暗号化されたプログラムの提供が可能となる。

【0185】請求項11に記載の情報処理装置および請求項12に記載の情報処理方法によれば、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものが、第2のキーを用いて復号化され、そ

の復号化された第1のキーを用いて、プログラムに付されている署名が正当なものであることが確認される。そして、署名が正当なものであることが確認された場合のみ、プログラムが実行される。また、請求項13に記載の記録媒体には、コンピュータに、プログラムに付されている署名を確認するときに用いる第1のキーを暗号化したものを、第2のキーを用いて復号化させ、その復号化された第1のキーを用いて、プログラムに付されている署名が正当なものであることを確認させ、署名が正当なものであることが確認された場合のみ、プログラムを実行させるためのプログラムが記録されている。従って、第1のキーが復号化され、さらに、プログラムに付されている署名が正当なものである場合のみ、そのプログラムが実行されるようにすることが可能となる。

【0186】請求項14に記載の情報処理装置および請求項15に記載の情報処理方法によれば、請求項11に記載の情報処理装置において署名が正当なものであると確認されるように、プログラムが処理される。また、請求項16に記載の記録媒体には、請求項11に記載の情報処理装置において署名が正当なものであると確認されるように処理されたプログラムが記録されている。従って、請求項11に記載の情報処理装置で実行可能なように処理されたプログラムの提供が可能となる。

【図面の簡単な説明】

【図1】 計算機1の資源と、そこに実装されたJava仮想マシン11の資源との対応関係を示す図である。

【図2】 Java仮想マシン11の処理を説明するための図である。

【図3】 Java仮想マシン11の処理を説明するための図である。

【図4】 Java仮想マシン11の処理を説明するための図である。

【図5】 本発明を適用したプログラム提供システムの一実施の形態の構成例を示すブロック図である。

【図6】 図5のソフトウェア開発者サーバ31の構成例を示すブロック図である。

【図7】 図5のプログラム認証機関サーバ32の構成例を示すブロック図である。

【図8】 図5のユーザ端末33の構成例を示すブロック図である。

【図9】 暗号化／復号化システムの構成例を示すブロック図である。

【図10】 デジタルサインを用いた暗号化／復号化システムの構成例を示すブロック図である。

【図11】 ソフトウェア開発者サーバ31の処理を説明するためのフローチャートである。

【図12】 ソフトウェア開発者サーバ31の処理を説明するためのフローチャートである。

【図13】 プログラム認証機関サーバ32の処理を説明するためのフローチャートである。

【図14】 プログラム実行システムの第1の機能的構成例を示すブロック図である。

【図15】 プログラム実行システムの第2の機能的構成例を示すブロック図である。

【図16】 ソフトウェア開発者サーバ31の処理を説明するためのフローチャートである。

【図17】 プログラム実行システムの第3の機能的構成例を示すブロック図である。

【図18】 プログラム認証機関サーバ32の処理を説明するためのフローチャートである。

【図19】 ロータの機能的構成例を示すブロック図である。

【図20】 プログラム実行システムの第4の機能的構成例を示すブロック図である。

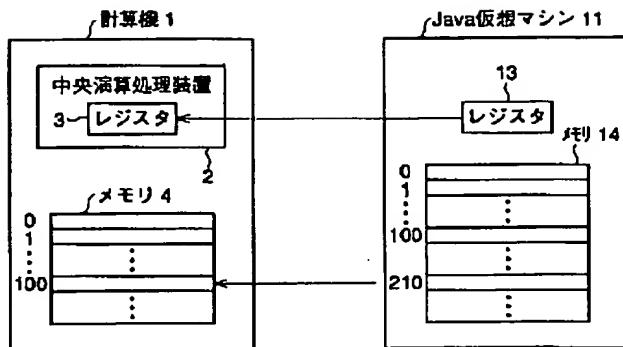
【図21】 ソフトウェア開発者サーバ31の機能的構成例を示すブロック図である。

【図22】 プログラム実行システムの第5の機能的構成例を示すブロック図である。

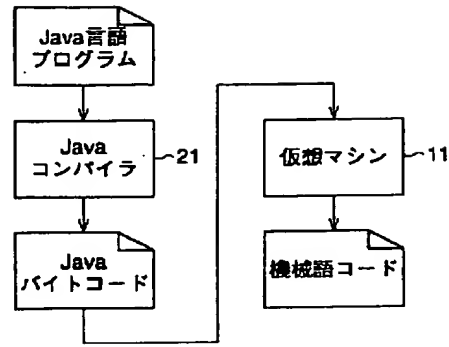
【符号の説明】

1 計算機, 2 中央演算処理装置, 3 レジスタ, 4 メモリ, 11 Java仮想マシン, 13 レジスタ, 14 メモリ, 21 Javaコンパイラ, 31 ソフトウェア開発者サーバ, 32 プログラム認証機関サーバ, 33 ユーザ端末, 34 ネットワーク, 35 記録媒体, 41 CPU, 42 ROM, 43 RAM, 44 入力部, 45 出力部, 46 補助記憶装置, 47 通信制御部, 51 CPU, 52 ROM, 53 RAM, 54 入力部, 55 出力部, 56 補助記憶装置, 57 通信制御部, 61 CPU, 62 ROM, 63 RAM, 64 入力部, 65 出力部, 66 補助記憶装置, 67 通信制御部, 71 暗号化器, 72 復号化器, 81 入力部, 82 復号部, 83 Java仮想マシン, 84 復号部, 91 ダイジェスト作成器, 92 暗号化器, 93 復号化器, 94 ダイジェスト作成器, 95 署名確認器, 101 入力部, 102 メッセージダイジェストシステム, 103 署名確認部, 104 仮想マシン入力制御部, 111 入力部, 112 メッセージダイジェストシステム, 113 署名確認部, 114 実行システム, 121, 122 暗号化器, 131, 132 復号部

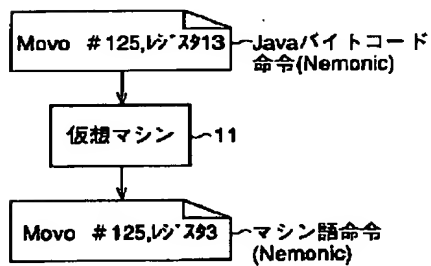
【図1】



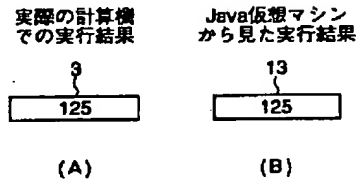
【図2】



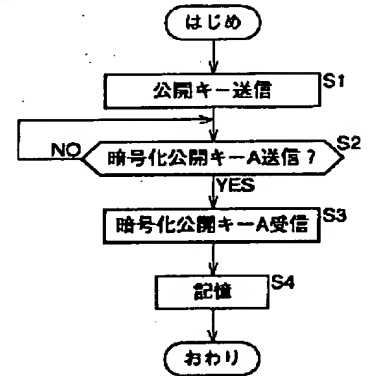
【図3】



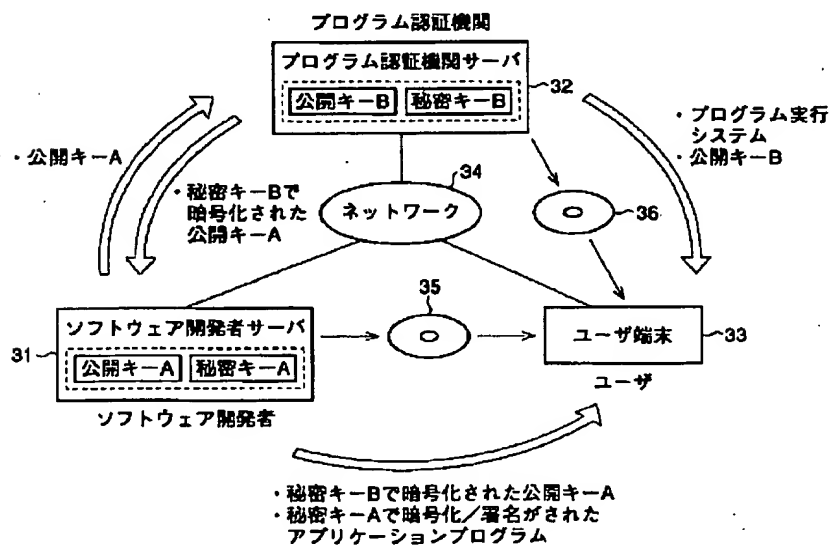
【図4】



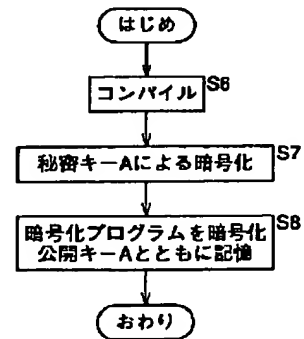
【図11】



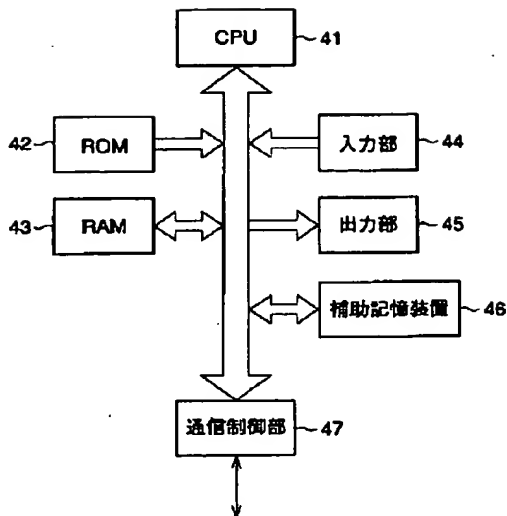
【図5】



【図12】

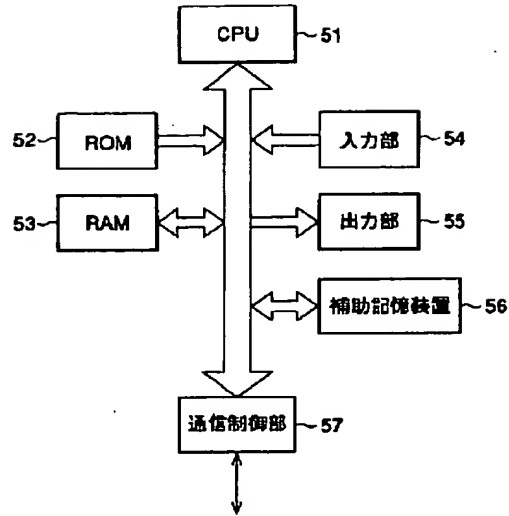


【図6】



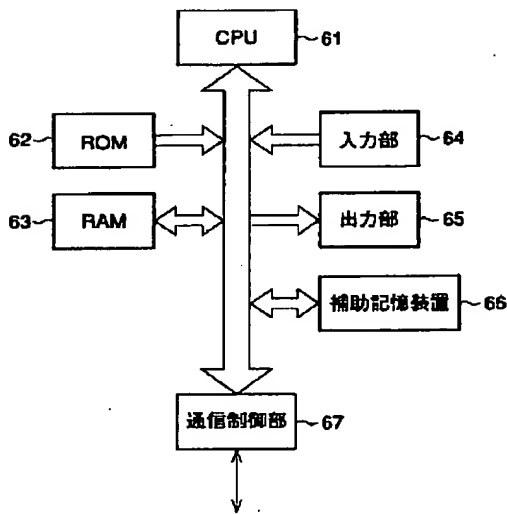
ソフトウェア開発者サーバ 31

【図7】



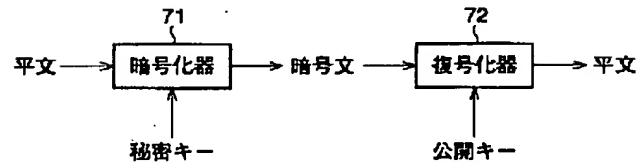
プログラム認証機関サーバ 32

【図8】



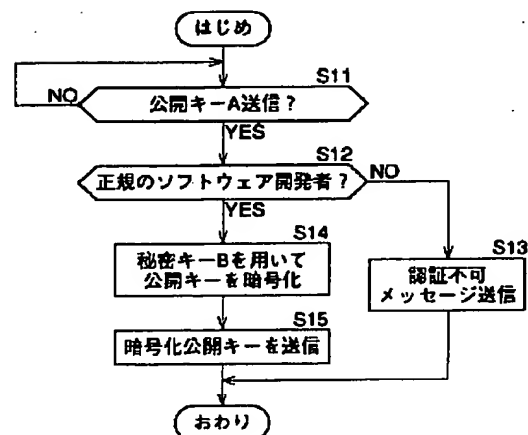
ユーザ端末 33

【図9】

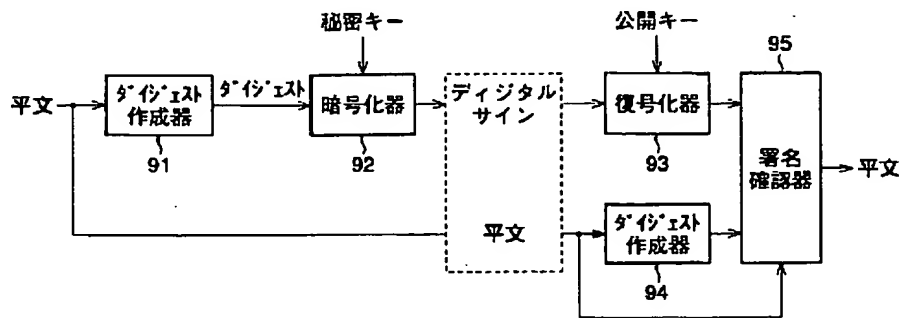


公開キー暗号化/復号化システム

【図13】

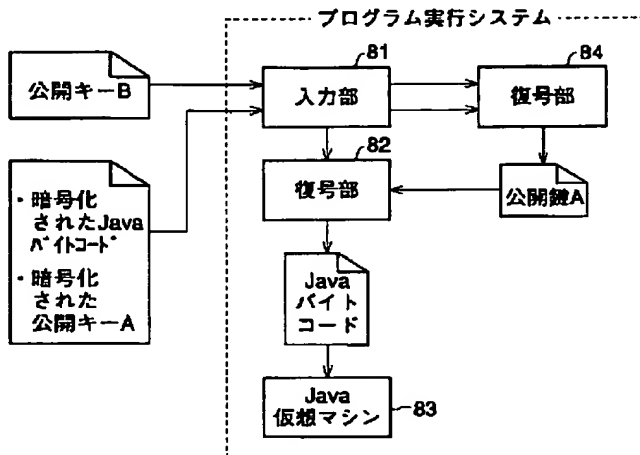


【図10】

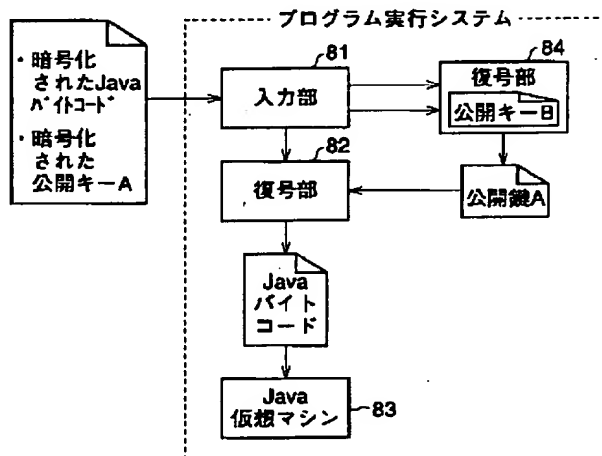


デジタルサインを用いた暗号化/復号化システム

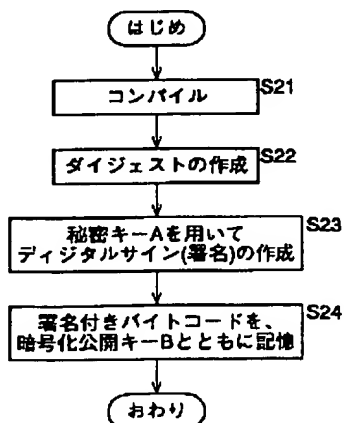
【図14】



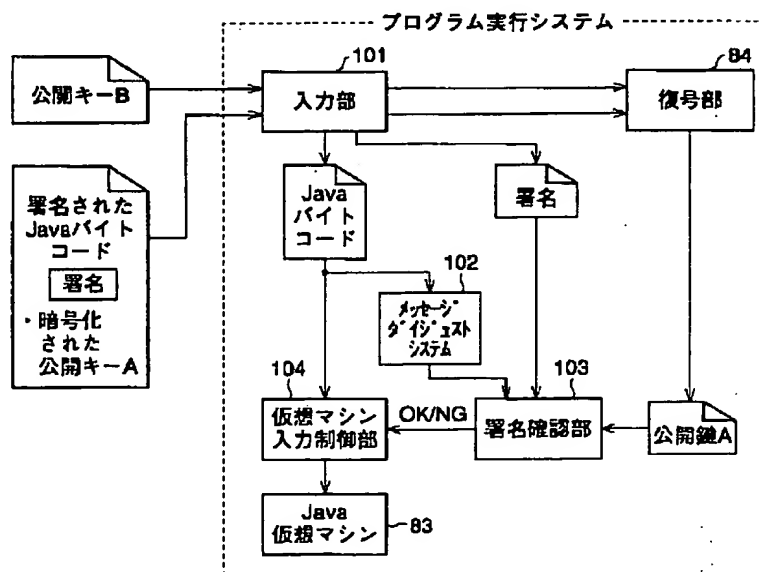
【図15】



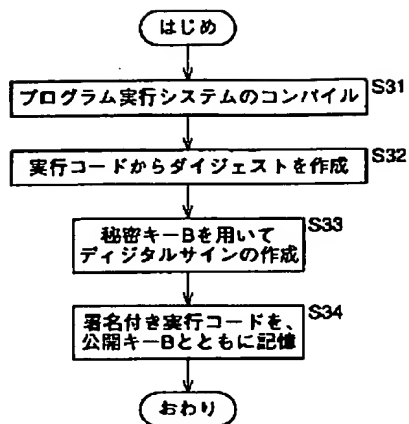
【図16】



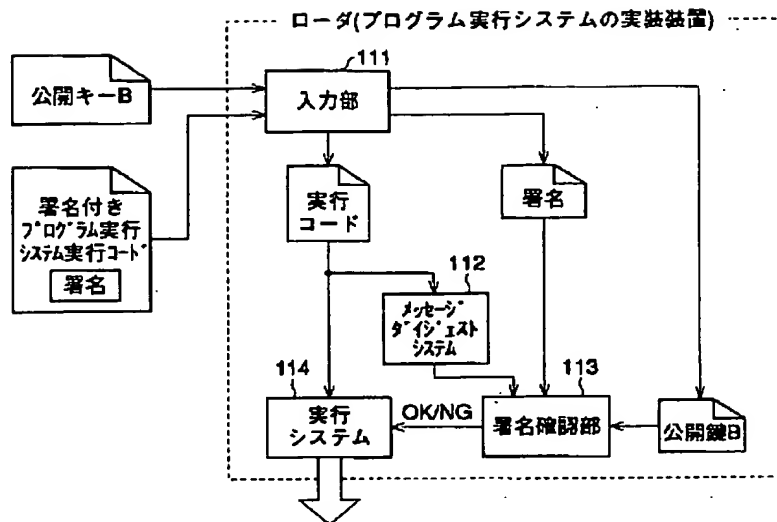
【図17】



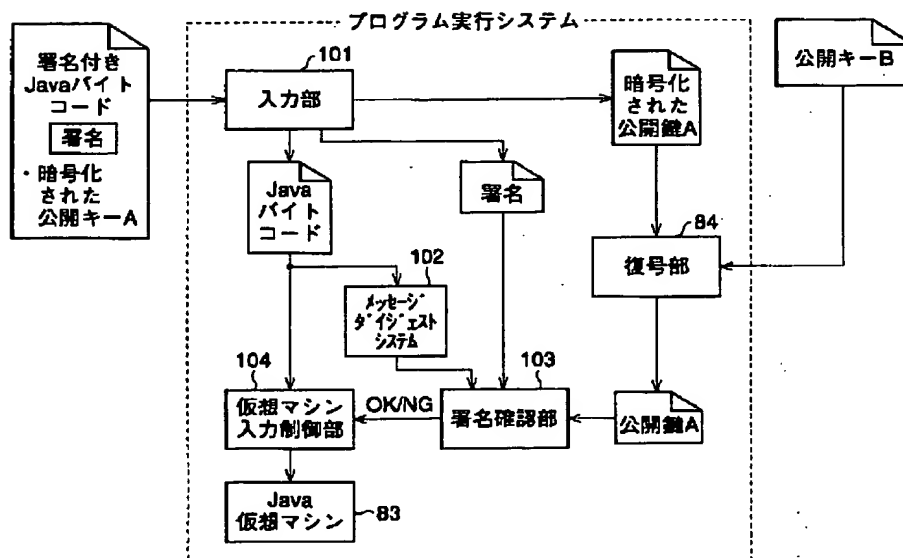
【図18】



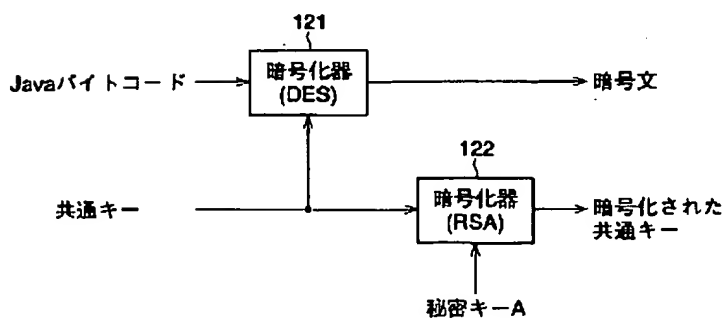
【図19】



【図20】



【図21】



【図22】

